A Quantitative Model for Analysis and Evaluation of Tor Hidden Service Discovery

Peipeng Liu, Xiao Wang, Xin He, Chenglong Li,
Shoufeng Cao, Longtao He, and Jiawei Zhu(✉)

National Computer Network Emergency Response Technical Team/Coordination Center, Beijing, China
zhujw.happy@163.com

Abstract. Tor is one of the most popular anonymous communication systems, and its ability of providing receiver anonymity makes *hidden services* more and more attractive. However, with the exposure of illegal contents such as child pornography and drug trades in hidden services, it becomes urgent to make a comprehensive analysis and evaluation of hidden services in the Tor network. In this paper, based on the frequent updates of hidden service descriptors, we proposed an approach to model Tor hidden service discovery as a generalized coupon collector problem with group drawings. Our experiments based on the real Tor network proved the efficiency and feasibility of the proposed model, which proved the possibility of harvesting most of hidden services with a small amount of resources.

Keywords: Tor · Hidden service · Discovery · Coupon collector

1 Introduction

Tor [1] is one of the most popular low-latency anonymous communication systems. Based on globally distributed volunteer-run relays, Tor uses several hops to forward users' messages in a layered-encryption way to prevent an external or internal attacker from correlating the two parties of a communication. Up to November 2016, there are about 7000 running Tor relays and 2000 bridges distributed around the world, and about 2 million users around the world using Tor to protect their communication anonymity [2].

In addition to protecting the client's anonymity, in order to hide the identities of service providers while enabling them to run normal web services, *hidden service* [3] was introduced into Tor in 2004. Recently, with the appearance of illegal contents such as child pornography and drug trades in hidden services, it becomes attractive to realize the current situation of hidden services in Tor network, such as their total number, content distribution, individual popularity and so on. All of these makes hidden service discovery a prerequisite.

In this paper, we model the hidden service discovery by running HSDirs as a generalized coupon collector problem, and based on the proposed model, we

G. Sun and S. Liu (Eds.): ADHIP 2017, LNICST 219, pp. 70–77, 2018.
https://doi.org/10.1007/978-3-319-73317-3_10

efficiently quantify the relationship between what you have and what you get in terms of hidden service discovery. Contributions in this paper can be summarized as follows:

1. We model the hidden service discovery by modeling HSDirs as a coupon collector problem with group drawings.
2. We quantify the relationship between consumed resources and collected hidden services in the discovery of Tor hidden services, and proved the feasibility of harvesting most of hidden services with a small amount of resources.

Rest of the paper is organized as follows, Sect. 2 introduced the background of Tor and hidden service, and simply summarized previous related work. Section 3 described our approach to model hidden service discovery as coupon collector problem. Section 4 presented our experiment strategies to evaluate the proposed model. Section 5 gave some discussions and Sect. 5 finally concluded the paper.

2 Background

2.1 Tor

Tor, the second-generation onion router [1], is a protocol that intends to anonymize network traffic in a low latency manner. Messages in Tor are forwarded through a multi-hop circuit in a layered-encryption manner, thus preventing a single attacker from knowing both parties of a communication.

With the introduction of *Hidden Service* in 2004 [1], Tor makes it possible for users to hide their locations while offering various kinds of web services, such as web publishing, instant messaging servers and so on. And other Tor users can connect to these hidden services without knowing providers' network identity.

2.2 Related Work

The receiver privacy provided by hidden service attracts more and more users to host web services in Tor network. The rapid growth of Tor hidden services makes it a hot topic in the anonymous research and lots of work have appeared to analyze and evaluate it.

Recently, the exposure of illegal hidden services [4] makes it appealing to analyze the size and content of Tor hidden services. George Kadianakis et al. added a statistics to the Tor software which can report the number of unique .onion addresses observed by a hidden service directory. And then based on these statistics, they extrapolated the total number of .onion addresses. Aiming to enumerate all Tor hidden services, Biryukov et al. described an efficient approach in [5]. They collected hidden services by deploying enough HSDirs based on a *shadowing* technique.

As Tor network grows and more stringent requirements on becoming HSDirs, we will inevitably have to face the predicament of using a small amount of resources to collect hidden services. That is, we have to study these issues:

1. How many hidden services can we discover if we don't have enough HSDir.
2. How many HSDirs are needed if we just want to collect a certain percentage of hidden services.

In this paper, we will present an model to describe the hidden service discovery by running HSDirs, and based on the model, we will quantify the relationship between consumed resources and collected hidden services, and finally give answers to the above questions.

3 Our Approach

In order to quantify the relationship between resources and collected hidden services, we model the discovery of hidden services by running HSDirs as a coupon collector problem. In this section, we will first introduce the coupon collector problem, and then describe our modeling method, and finally deduce the formulas to calculate two key parameters in the model.

3.1 Coupon Collector Problem

In the classical coupon collector problem, all n coupons are obtained with an equal chance of $1/n$. To collect all different coupons, the collector needs to do $\Theta(n \ln n)$ samples on average. In [6], Stadje extended the classical coupon problem to the situation where samples are done with replacement of equiprobable groups of a fixed size g. And in this situation, given a subset $A \subset S$ (S is the set of all different coupons), Stadje deduced the distributions of the number of distinct elements of A after k samples and the sample size necessary to obtain at least say x elements of A. In this paper, we will analyze and evaluate the discovery of hidden services by running HSDirs based on the extended coupon collector problem.

3.2 Discover Hidden Services by Running HSDirs

According to the protocol of Tor hidden service, a hidden service has to publish its descriptors to several HSDirs before can be accessed by any user.Tor relays with the 'HSDir'.flag forms a distributed hash table to store the descriptors published by hidden services. Once descriptor identifiers are determined, the hidden service first arranges HSDirs using their fingerprints in a closed fingerprint circle and then chooses the three closest HSDirs in positive direction (fingerprint values of them are greater than the descriptor identifiers of the hidden service). As a hidden service generates and publishes two replicas of descriptors by default, 2 sets of 3 HSDirs with consecutive fingerprints are chosen to store corresponding descriptors.

It's worth to note that, as each hidden service changes its descriptor identifiers every 24 h, and thus probably changing its responsible HSDirs. A particular HSDir will get an opportunity to discover a particular hidden service whenever the hidden service changes its descriptor identifiers. And this makes it possible to run a few HSDirs to collect more different hidden services over time.

3.3 Modeling

We model the discovery of hidden services by running HSDirs as the extended coupon collector problem with group drawings. We collect a group of hidden services by running several HSDirs everyday, aiming at collecting as many hidden services as possible. Assume the total number of Tor hidden service is S, and the number of hidden services collected by h HSDirs one day is g. Then we can map the discovery of hidden service to coupon collector problem as Table 1.

Table 1. Modeling

Symbols	Coupon collector	Hidden service discovery
S	Total number of coupons	Total number of hidden services
k	Number of samples	Number of days (24 h)
g	Size of coupon group	Number of hidden services collected one day (by h HSDirs)

As shown in Table 1, we take the total number of hidden services as the number of coupons, and due to the specification of Tor hidden service protocol (i.e., hidden service changes its descriptor identifiers every 24 h), we set the sampling interval to one day. Finally, the number of hidden services collected by our h HSDirs in one day corresponds to the size of coupon group in [6].

However, before we can use the conclusion of coupon problem to quantify the relationship between consumed resources and collected hidden services, we have to first specify the values of S and g.

Total number of Tor hidden service. According to the design of Tor hidden service [7], each hidden service generates 2 descriptors with different identifiers, and each descriptor chooses 3 responsible hidden service directories to publish. As both the descriptor identifiers and the fingerprints of HSDirs are generated by SHA1 function, it's reasonable to assume that the probability a descriptor is published to each HSDir is equal[1]. Given the total number of HSDirs N which can be learn from consensus files, the probability that a HSDir receives a particular descriptor is $3/N$, because once the descriptor identifier falls in one of the three intervals before the HSDir in the fingerprint circular, this descriptor will choose the HSDir as one of its responsible HSDirs. Besides, as each hidden service updates its descriptors once per 24 h and generate two descriptors each time, an HSDir thus gets 2 chances to be chosen by one hidden service in one day. According to the Bernoulli trial [9], the probability for a HSDir to store a given hidden service can be calculated as:

$$q = 1 - (1 - 3/N)^2 = 6/N - 9/N^2 \tag{1}$$

[1] In [8], George Kadianakis et al. computed the fraction of descriptors that a HSDir is responsible for, and their results showed that the fraction value is very small (0.024%), and there is little difference for this value between different HSDirs.

Thus, given the average number of hidden services collected by an HSDir one day, the total number of hidden services can be estimated by dividing the average number with the above probability.

Size of coupon group. Another parameter in the extended coupon collector problem is the size of coupon group, and in this section, we will estimate the number of hidden services collected by h HSDirs per 24 h, i.e., the size of the coupon group in one sample in the extended coupon collector problem. At first, we make the following definition:

Catch Probability: The probability that a hidden service chooses at least one of the deployed h HSDirs as its responding HSDirs, denoted by p.

At one hand, p is affected by the number of deployed HSDirs. Assume there are h deployed HSDirs and totally N HSDirs. As the fingerprints of the deployed HSDirs can be carefully chosen so that distances of any two deployed HSDirs is larger than 3 in the fingerprint circular. As a result, no descriptor can be published to more than one of the deployed HSDirs. However, with two different descriptors, it's possible for a hidden service to be published to two different deployed HSDirs. Due to the fact that each hidden service updates its descriptors once per 24 h and generate two descriptors each time, a hidden service has two chances to select the deployed HSDirs in one day. As a result, the probability that a hidden service chooses at least one of the h HSDirs as its responsible HSDirs can be given by:

$$1 - (N - 3h)^2/N^2 \tag{2}$$

where $(N - 3h)^2/N^2$ is the probability that neither of the two descriptors of a hidden service chooses one of the deployed h HSDirs as their corresponding HSDirs. Thus, the number of hidden services collected by h HSDirs one day can be estimated by $S * p$.

Once the total number of hidden services and the size of group are known, combining with the conclusions in [6], we can get the expectation of the number of distinct hidden services collected after k days with h HSDirs:

$$E(X_k(S)) = S\left[1 - \left(1 - \frac{g}{S}\right)^k\right] \tag{3}$$

where, S is the total number of hidden services by Formula 1 and g is the number of hidden services collected by h HSDirs one day by $S * p$.

4 Evaluation

We have deployed several HSDirs to collect data to validate our model in early 2015. In this section, we will present our experiments and evaluation, and the results proved the efficiency of our model.

4.1 Experiment Deployment

Four machines have been deployed with 3 locating in American and 1 in Japan, and by configuring 2 Tor instances on each machine, 8 HSDirs are finally operated. We further configured the fingerprints of deployed HSDirs not consecutive so that a same hidden service descriptor would not be published to any two HSDirs we ran. At last, by modifying Tor's source code as [10], we recorded the number of hidden service .onion addresses published to the 8 HSDirs (for the privacy issues, we didn't record the actual .onion addresses).

4.2 Evaluation

By default, hidden services change their responsible HSDirs every 24 h (note that not all hidden services are synchronized, and thus different hidden services may update their descriptor identifiers at different times). We recorded the number of collected hidden services by each deployed HSDir per 24 h, as this makes all hidden services have one chance to update its two descriptors (and thus two chances to choose our HSDirs as its responsible HSDirs) during each statistical period, which matches the requirements of coupon collect problem, i.e., each coupon is collected by same probability in a sample. Figure 1 shows the number of total collected hidden services by each HSDir per day respectively.

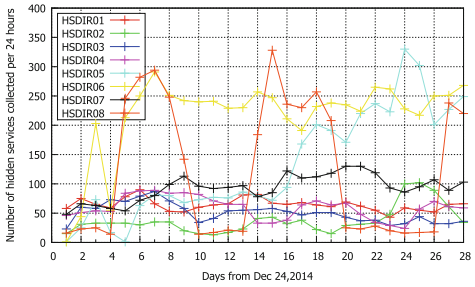

Fig. 1. Hidden services collected by each HSDir per day

Finally, 13337 distinct hidden services are collected by the 8 HSDirs in 28 days. And thus on average $13337/(28 * 8) = 59.54$ hidden services are collected by each HSDir per day. It's worth to note that the data collected by HSDir5, HSDir6 and HSDir8 is not as stable as the rest 5 HSDirs, and we are still working to find the reasons. When we exclude the data by these three HSDirs, the average hidden services collected by each HSDir per day is $8046/(28*5) = 57.47$, which is almost the same with the previous 59.54. For simplicity, we take 60 as the average number of hidden services collected by one HSDir per day in the following.

Combing with probability Eq. 1, the total number of hidden services is estimated as $60/[1-(1-3/2935)^2] = 29365$, where 2935 is the number of HSDirs in

Tor network got from consensus files, and this result coincides with the statistic (about 30000) announced on Tor network [11] at the writing time.

It's also necessary to evaluate the size of the group of hidden services collected one day to model the hidden service collecting problem as coupon collector problem. As analyzed in Sect. 3.3, we can collect $29365 * (1-(2935-3*8)^2/2935^2) = 478$ hidden services one day by 8 HSDirs given the total number of hidden services is 29365.

Figure 2 gives the theoretical and the experimental number of accumulated discovered hidden services, where the theoretical value is calculated by Eq. 3 with S set 29365 and g set to 478, while the experimental value is calculated by summing the number of distinct hidden services collected by 8 HSDirs. The result indicates that the proposed model in this paper is consistent with the discovery of hidden service by running HSDirs with a high degree.

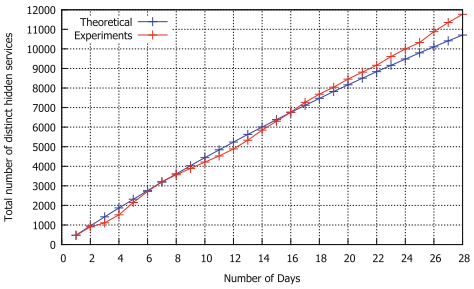

Fig. 2. Theoretical vs experiment

Given the efficiency of the proposed model, we can quantify the relationship between the collected hidden services and the consumed resources, i.e., number of HSDirs and number of days, as illustrated in Fig. 3. Specifically, according to the proposed model, with the estimation that there are 29365 hidden services in total and a HSDir can discover 60 hidden service one day, when 50 EC2 (the

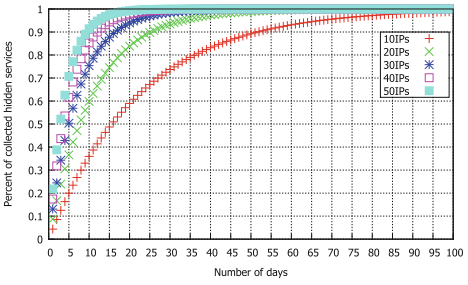

Fig. 3. Conjecture

resources needed to harvest hidden services in [5]) instances are deployed, more than 90% hidden services will be discovered after 10 days.

5 Conclusion

In this paper, we proposed a model based on coupon collector problem to describe Tor hidden services discovery by running HSDirs. The proposed model can efficiently quantify the relationship between consumed resources and collected hidden services. Experiments based on the real Tor network proved the efficiency and feasibility of the proposed model, and can be used to guide the harvesting of hidden services in Tor network with a small amount of resources.

Acknowledgments. This research is funded by National Key Research & Development Plan of China under Grant 2016YFB0801200, 2016YFB0801602 and 2016QY05X1000.

References

1. Dingledine, R., Mathewson, N., Syverson, P.: Tor: The second-generation onion router. Technical Report, DTIC Document (2004)
2. http://metrics.torproject.org
3. https://www.torproject.org/docs/hidden-services.html.en
4. http://en.wikipedia.org/wiki/SilkRoad(marketplace)
5. Biryukov, A., Pustogarov, I., Weinmann, R.: Trawling for Tor hidden services: detection, measurement, deanonymization. In: 2013 IEEE Symposium on Security and Privacy (SP), pp. 80–94. IEEE (2013)
6. Stadje, W.: The collector's problem with group drawings. Adv. Appl. Probab. **22**, 866–882 (1990)
7. https://gitweb.torproject.org/torspec.git/plain/rend-spec.txt
8. https://research.torproject.org/techreports/extrapolating-hidserv-stats-2015-01-31.pdf
9. http://en.wikipedia.org/wiki/Bernoulli_trial
10. https://github.com/DonnchaC/tor/
11. https://metrics.torproject.org/hidserv-dir-onions-seen.html